Quick Start: µC/OS-II on Microsemi's SmartFusion2 Development Kit

Required Hardware

This quick-start guide provides step-by-step instructions for running Micriµm's µC/OS-II kernel on the SmartFusion2 Development Kit from Microsemi. The example project described herein is intended to be used in conjunction with the demo design that is provided with the kit. (http://www.microsemi.com/fpga-soc/design-resources/dev-kits/smartfusion2/smartfusion2-development-kit.) µC/OS-II is a highly portable kernel and could easily be adapted to other SmartFusion2-based boards and FPGA designs, but this document focuses solely on the Development Kit and its demo design. If you are interested in a different hardware platform, you can contact Micriµm for additional information.

The SmartFusion2 Development Kit

Required Software

The instructions provided in this guide are for Microsemi's SoftConsole IDE. A free, Eclipse-based IDE, SoftConsole is part of the Libero SoC toolset that can be downloaded from Microsemi's Web site: http://www.microsemi.com/fpga-soc/design-resources/design-software/libero-soc. (Libero must be licensed, but a free Gold license, supporting a number of SmartFusion2 devices, including the M2S050 featured on the Development Kit board, can be obtained from Microsemi at no cost.) v11.1 of Libero SoC was used in testing the example project covered by this guide. However, Micriµm's software is highly flexible with regard to tool-chains and IDEs. If you're planning to run µC/OS-II under an environment differing from the one described in this document, you can contact Micriµm for additional information.


The SoftConsole IDE
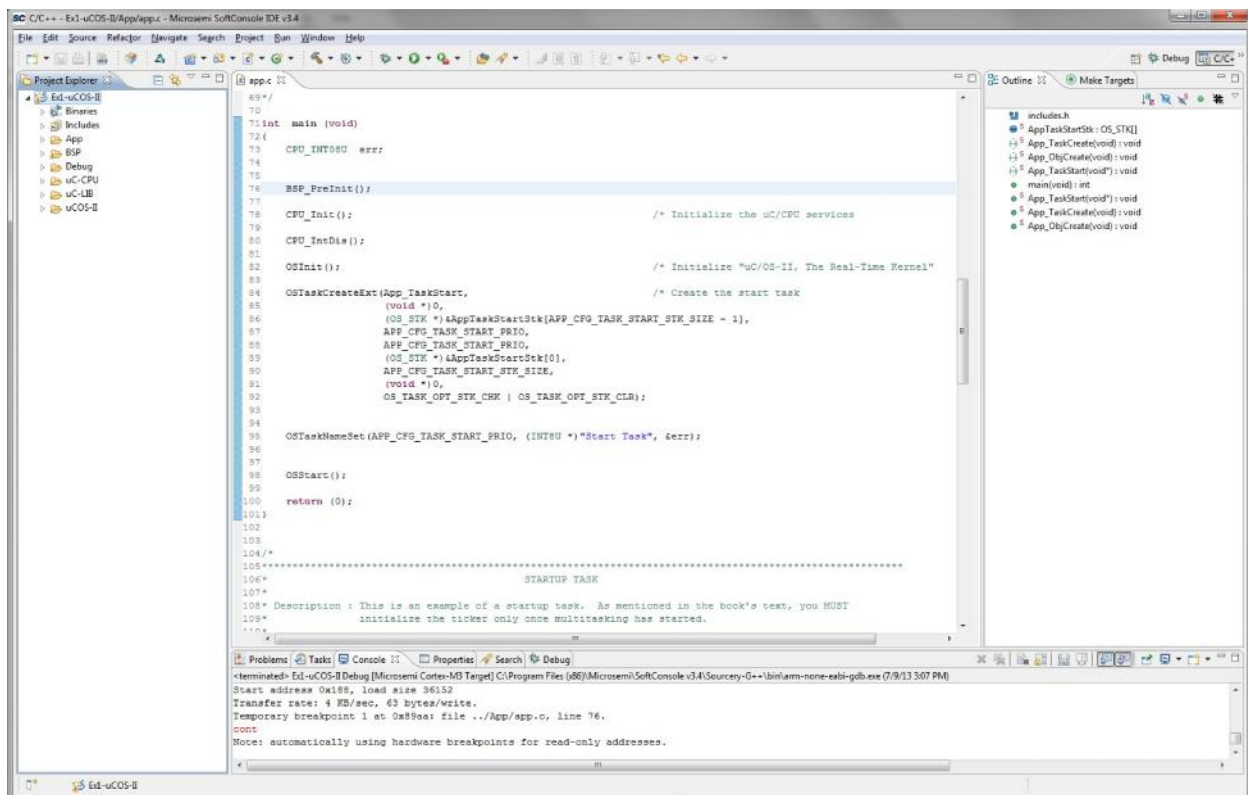
Building and Running the Example Project

1. The µC/OS-II example project covered in the following instructions is provided in a zip file named *uCOS-II-SmartFusion2.zip*. Once you've downloaded this file from Micriµm's Web site (http://micrium.com/downloadcenter/download-results/?searchterm=hm-microsemi&supported=true), you should extract its contents. Since compilers and other tools sometimes have problems with long path names, it is recommended that you place the zip file's extracted contents close to your root folder.

2. In order to run the µC/OS-II example project, you'll need to have Microsemi's demo design for the SmartFusion2 Development Kit programmed onto your board. The *SmartFusion2 Development Kit Demo Design User Guide* provides instructions for programming the design. This document is available from the Microsemi Web site: http://www.microsemi.com/fpga-soc/design-resources/dev-kits/smartfusion2/smartfusion2-development-kit.

3. You'll download the executable code for the µC/OS-II example project using the FlashPro4 programmer provided with the Development Kit. If you have not already connected your board to your PC via the FlashPro4, you should do so now. You should also make sure that power has been applied to the board.

4. You should now open SoftConsole, using the IDE's entry in the **Start** menu. Like most Eclipse-based IDEs, SoftConsole typically asks for a workspace location before it begins running. To ensure that the µC/OS-II example project will build correctly, you should specify the below location. (*<Micrium Install>* represents the folder into which you placed the contents of *uCOS-II-SmartFusion2.zip*.)

   *<Micrium Install>\Micrium\Software\EvalBoards\Microsemi\SF2-Dev-Kit\SC*

5. The µC/OS-II example project is composed of a number of files that are referenced through a path variable named `MICRIUM_FOLDER`. Thus, before you actually import the project into your workspace, you should define this variable. You should begin by selecting **Preferences** from the **Window** menu, and then, in the list on the left-hand side of the resulting dialog, browsing to **General>Workspace>Linked Resources**. You should subsequently see the dialog content shown on the next page of this document.
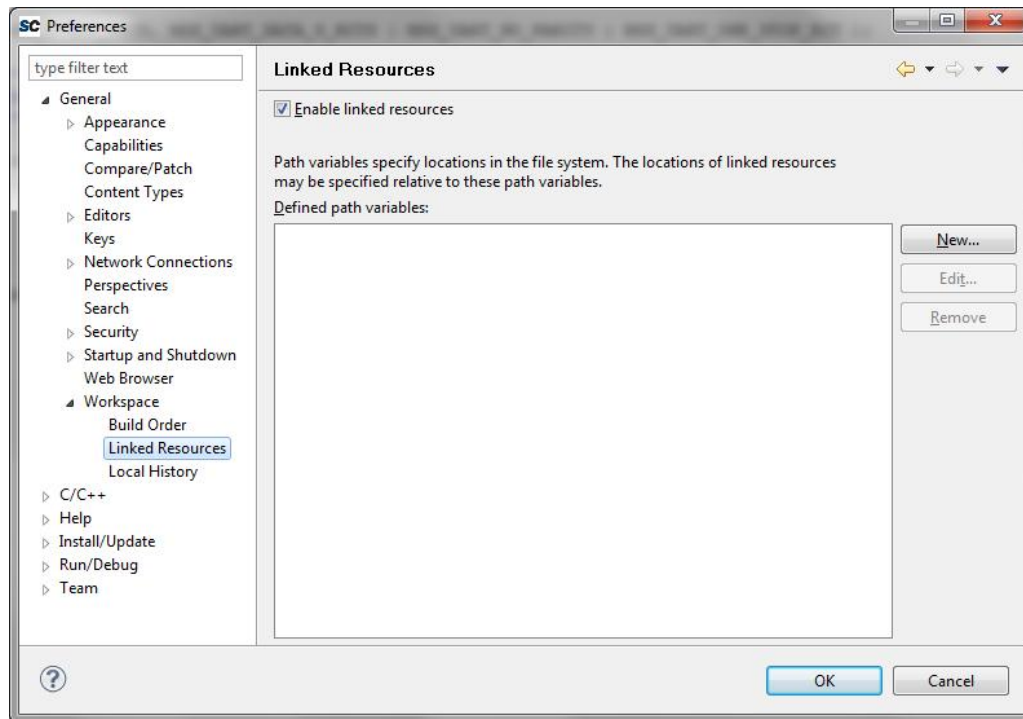
6. The **New…** button appearing on the **Linked Resources** page will allow you to specify a path variable. You should now click this button. In the ensuing **New Variable** dialog, you should enter `MICRIUM_FOLDER` as your new variable's **Name**, and you should provide the below path as the variable's **Location**. You can then click the **OK** button in both the **New Variable** and **Preferences** dialogs.

   <Micrium Install>\Micrium

7. With `MICRIUM_FOLDER` added to your workspace's list of path variables, you should now begin importing the example project by selecting **Import** from the **File** menu. On the first page of the **Import** dialog, you should select **General>Existing Projects into Workspace**, as shown in the screenshot on the following page of this document. You should then click the **Next** button.

Existing Projects into Workspace
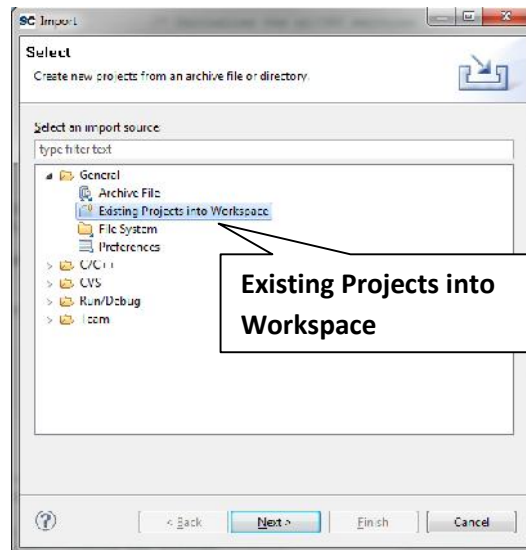
8. On the second page of the **Import** dialog, you should confirm that **Select root directory** has been chosen, and you should then click the **Browse** button corresponding to this option. The folder to which you should browse before clicking **OK** is listed below. After specifying this folder, you can close the **Import** dialog by clicking its **Finish** button.

   *<Micrium Install>\Micrium\Software\EvalBoards\Microsemi\SF2-Dev-Kit\SC\Ex1-uCOS-II*

9. The example project should now appear in your **Project Explorer**, as shown below. You should attempt to build the project by right-clicking its name and selecting **Build Project** from the resulting menu.
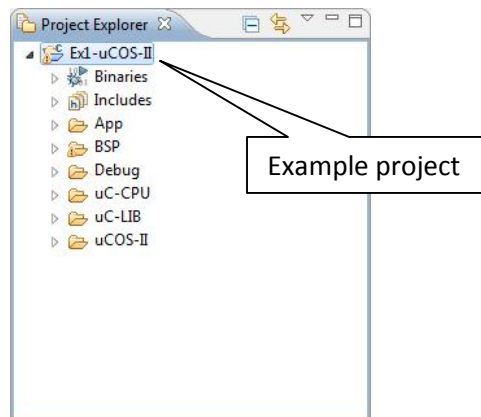
Example project

10. The **Console,** normally appearing at the bottom of the main SoftConsole window, is responsible for showing any problems that take place while a project is being built. If the **Console** did not report any errors or warnings for the example project, you should now right-click the project and select **Debug As>Debug Configurations…** from the resulting menu.   The **Debug Configurations** dialog will then appear.

11. Within **Debug Configurations**, you should select **Microsemi Cortex-M3 Target** and then click the **New** button shown below. After making sure that the **Name** field for the new configuration is not empty, and confirming that the **Main** page resembles the one shown below, you should click the **Debug** button.
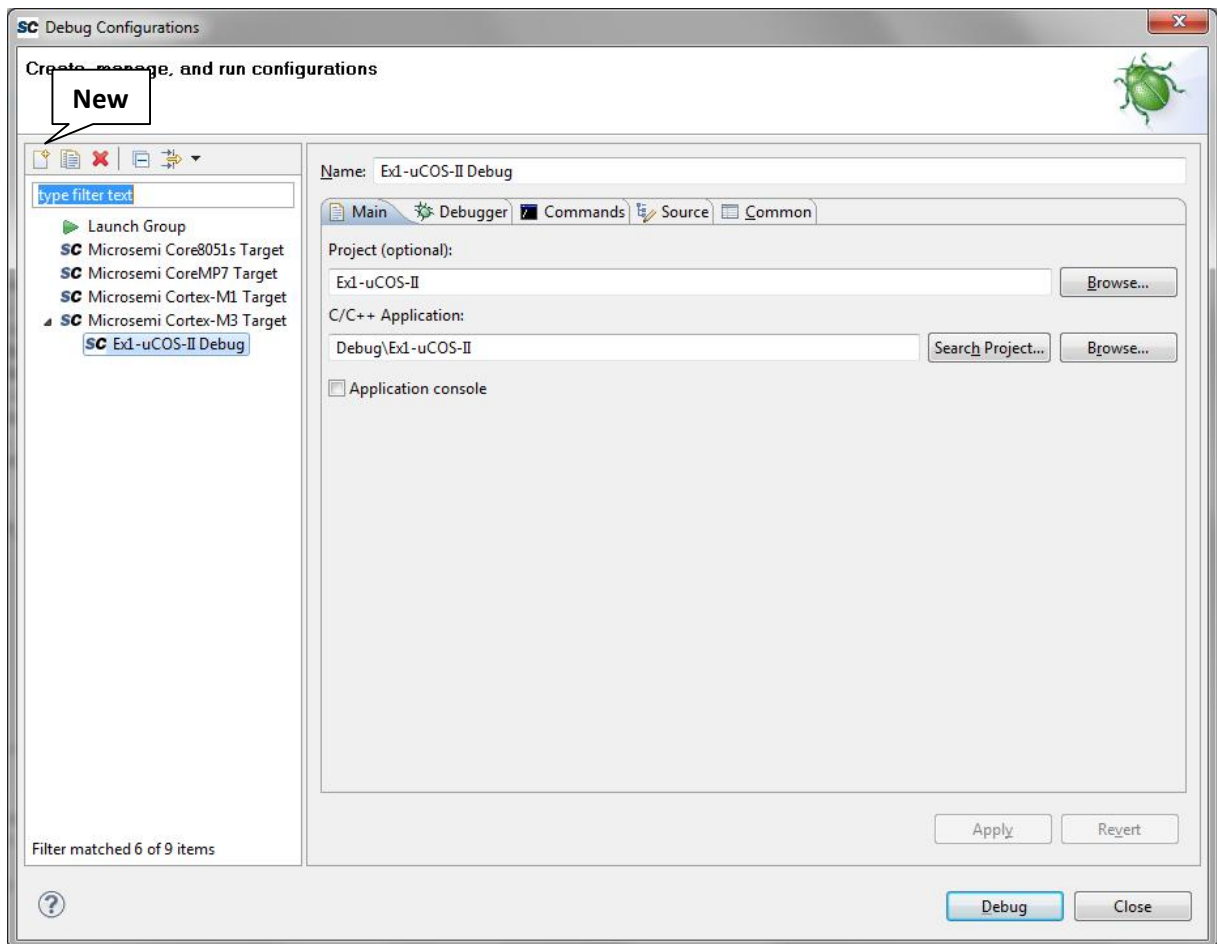
12. SoftConsole, like other Eclipse-based IDEs, incorporates multiple perspectives. After you click **Debug**, the below dialog, asking whether you'd like to switch from the C/C++ perspective to the Debug perspective, should appear. You should simply click **Yes**.
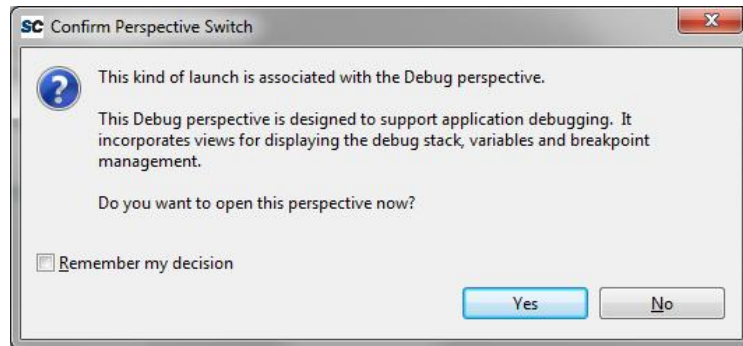
13. Once you enter the Debug perspective, you should be able to see the progress of SmartConsole's debugger in the **Console**. The debugger should ultimately download the example project's code and stop execution of this code at a breakpoint in `main()`. If you click the **Resume** button shown below, the code should begin running. You can check your board's LEDs to confirm that the code is running correctly. **LED2** and **LED3** should blink, while **LED1** and **LED6** should be continuously asserted.

Resume

Licensing

μC/OS-II is a source-available real-time kernel; it is <u>not</u> open source. Under the source-available model, which Micriμm pioneered, the kernel's full source code can be evaluated at no cost. This code can also be used free of charge in academic projects. Developers planning to use the code to develop a product, however, must purchase a license. Additional licensing information can be obtained from Micriμm; contact information is provided below.

Micriμm Contact Information

Phone: +1 954 217 2036
Fax: +1 954 217 2037
E-Mail: sales@micrium.com
URL: http://www.micrium.com